

Task – 19/Aug/25

Task:

create two ec2 machine

1st machine: install jenkins

2nd machine: install tomcat

write Jenkins pipeline which should be able to do ssh on 2nd machine and deploy addressbook app

Step 1. Creating 2 EC2 Machines

Note: Use t3.large as it has appropriate disk size to run Jenkins pipeline and is cost effective

a) Creating EC2 for Jenkins

Name and tags [Info](#)

Name
 [Add additional tags](#)

▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI contains the operating system, application server, and applications for your instance. If you don't see a suitable AMI below, use the search field or choose **Browse more AMIs**.

Recents

Quick Start

Amazon Linux
aws

macOS
Mac


Ubuntu
ubuntu

Windows
Microsoft

Red Hat
Red Hat

SUSE Linux
SUSE

Debian
debian


Browse more AMIs
Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type
ami-0704b375f44451515 (64 bit x86) / ami-0704b375f44451515 (64 bit x86)

Free tier eligible

Create key pair



Key pair name

Key pairs allow you to connect to your instance securely.

JK-Tomcat-kp

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type

☒ RSA

RSA encrypted private and public key pair

☐ ED25519

ED25519 encrypted private and public key pair

ED25519 encrypted private and public key pair

Private key file format

☒ .pem

For use with OpenSSH

☐ .ppk

For use with PuTTY



When prompted, store the private key in a secure and accessible location on your computer. **You will need it later to connect to your instance.** [Learn more](#)

Cancel

Create key pair

☐ Create security group

☒ Select existing security group

Common security groups [Info](#)

Select security groups

Pulkit-JK-SQ-SG sg-0592eb194a680d669 X
VPC: vpc-0047b25160c3ca412

[Compare security group rules](#)

Security groups that you add or remove here will be added to or removed from all your network interfaces.

▼ Configure storage [Info](#)

Advanced

1x 8 GiB gp3 Root volume, 3000 IOPS, Not encrypted

b) Creating EC2 for Tomcat

Name and tags [Info](#)


Name

Tomcat-ec2-task

[Add additional tags](#)

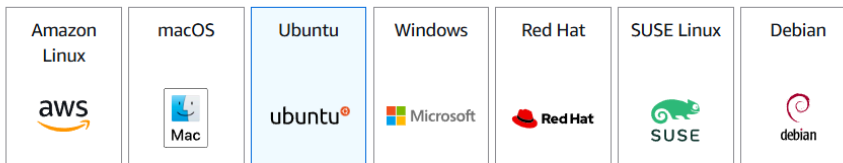
▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI contains the operating system, application server, and applications for your instance. If you don't see a suitable AMI below, use the search field or choose [Browse more AMIs](#).

 Search our full catalog including 1000s of application and OS images

Recents

[Quick Start](#)



[Browse more AMIs](#)

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type
ami-020cba7c55df1f615 (64-bit (x86)) / ami-07041441b708acbd6 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible

▼ Instance type [Info](#) | [Get advice](#)

Instance type

t3.small Free tier eligible
Family: t3 2 vCPU 2 GiB Memory Current generation: true
On-Demand SUSE base pricing: 0.0518 USD per Hour On-Demand Linux base pricing: 0.0208 USD per Hour
On-Demand RHEL base pricing: 0.0496 USD per Hour
On-Demand Ubuntu Pro base pricing: 0.0243 USD per Hour
On-Demand Windows base pricing: 0.0392 USD per Hour

☒ All generations

[Compare instance types](#)

[Additional costs apply for AMIs with pre-installed software](#)

▼ Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

Jk-Tomcat-kp

 [Create new key pair](#)

Firewall (security groups) [Info](#)


A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☐ Create security group

☒ Select existing security group

Common security groups [Info](#)

Select security groups

Pulkit-JK-SQ-SG sg-0592eb194a680d669 
VPC: vpc-0047b25160c3ca412

 [Compare security group rules](#)

Security groups that you add or remove here will be added to or removed from all your network interfaces.

▼ **Configure storage** [Info](#) Advanced

1x GiB Root volume, 3000 IOPS, Not encrypted

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage

[Add new volume](#)

Step 2. Installing Connecting to SSH

Connect [Info](#)
Connect to an instance using the browser-based client.

[EC2 Instance Connect](#) [Session Manager](#) [SSH client](#) [EC2 serial console](#)

Instance ID

Connection type

☒ Connect using a Public IP
Connect using a public IPv4 or IPv6 address

☐ Connect using a Private IP
Connect using a private IP address and a VPC endpoint

☒ Public IPv4 address

☐ IPv6 address
-

Username
Enter the username defined in the AMI used to launch the instance. If you didn't define a custom username, use the default username, ubuntu.

Note: In most cases, the default username, ubuntu, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

[Cancel](#) [Connect](#)

Step 3. Installing Java JDK in both EC2 as both Tomcat and Jenkins requires it

```
apt install -y openjdk-11-jdk
```

Note : Since both Jenkins and Tomcat are running on different EC2's thus we do not require to change connector port of any of them

Step 4. Installation of Jenkins and Tomcat

a) Installing Jenkins in EC2

- Running basic updates

To run a command as administrator (user "root"), use "sudo <command>". See "man sudo_root" for details.

```
ubuntu@ip-172-31-33-154:~$ sudo su
root@ip-172-31-33-154:/home/ubuntu# apt update
```

i-0f47d814cc045a32d (Jenkins-ec2-task)

PublicIPs: 98.80.65.113 PrivateIPs: 172.31.33.154

- Running Jenkins installation command

```
root@ip-172-31-33-154:/home/ubuntu# curl -fsSL https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key | sudo tee /usr/share/keyrings/jenkins-keyring.asc > /dev/null
echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] https://pkg.jenkins.io/debian-stable binary/ | sudo tee /etc/apt/sources.list.d/jenkins.list > /dev/null
sudo apt-get update -y
sudo apt-get install -y jenkins
```

- Installing maven

```
root@ip-172-31-33-154:/home/ubuntu# sudo apt-get install -y maven
```

i-0f47d814cc045a32d (Jenkins-ec2-task)

b) Installing Apache Tomcat

- Running basic updates

```
ubuntu@ip-172-31-34-39:~$ sudo su
root@ip-172-31-34-39:/home/ubuntu# apt update
```

i-0243e5919b7e02d81 (Tomcat-ec2-task)

PublicIPs: 98.81.126.197 PrivateIPs: 172.31.34.39

- Installing Tomcat

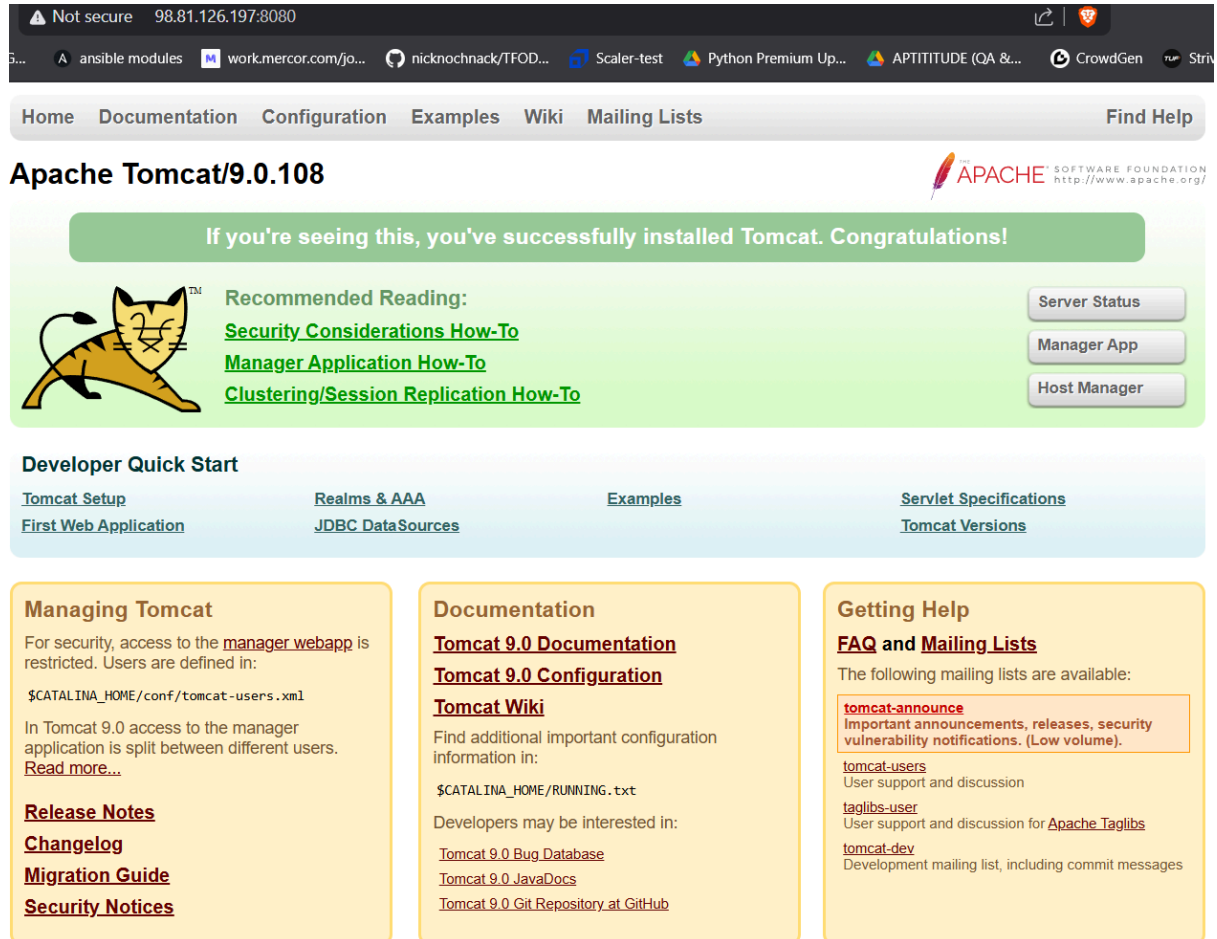
```
9 wget https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.108/bin/apache-tomcat-9.0.108.zip
0 unzip apache-tomcat-9.0.108.zip
```

c) Starting Tomcat (Going to bin in apache tomcat directory and running startup.sh file)

```
root@ip-172-31-34-39:/home/ubuntu# cd apache-tomcat-9.0.108/bin
root@ip-172-31-34-39:/home/ubuntu/apache-tomcat-9.0.108/bin# ls
bootstrap.jar catalina.sh commons-daemon-native.tar.gz configtest.sh digest.sh setclasspath.bat shutdown.sh tomcat-juli.jar tool-wrapper.sh
catalina-tasks.xml ciphers.bat commons-daemon.jar daemon.sh makebase.bat setclasspath.sh startup.bat tomcat-native.tar.gz version.bat
catalina.bat ciphers.sh configtest.bat digest.bat makebase.sh shutdown.bat startup.sh tool-wrapper.bat version.sh
```

```
root@ip-172-31-34-39:/home/ubuntu/apache-tomcat-9.0.108/bin# chmod +x *.sh
root@ip-172-31-34-39:/home/ubuntu/apache-tomcat-9.0.108/bin# ./startup.sh
```

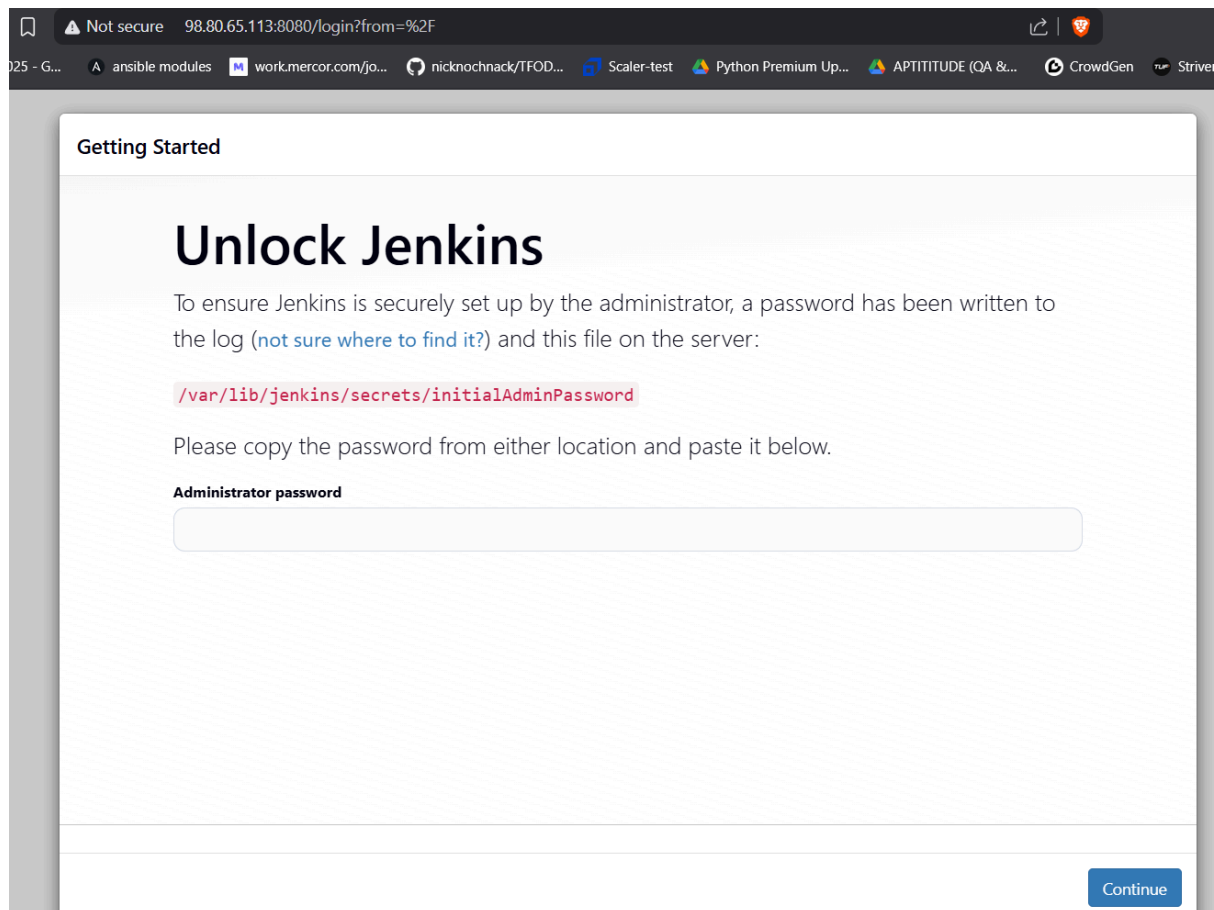
d) Checking Tomcat Installation on port 8080 (http://<Public_IP_address>:8080)



The screenshot shows the Apache Tomcat 9.0.108 web interface. The browser address bar shows the URL `http://98.81.126.197:8080`. The page has a navigation bar with links: Home, Documentation, Configuration, Examples, Wiki, Mailing Lists, and a Find Help button. The main content area features a green banner with the message: "If you're seeing this, you've successfully installed Tomcat. Congratulations!". Below this, there is a section for "Recommended Reading" with links to "Security Considerations How-To", "Manager Application How-To", and "Clustering/Session Replication How-To". To the right of these links are three buttons: "Server Status", "Manager App", and "Host Manager". Below the recommended reading section is a "Developer Quick Start" section with links to "Tomcat Setup", "First Web Application", "Realms & AAA", "JDBC DataSources", "Examples", "Servlet Specifications", and "Tomcat Versions". At the bottom, there are three yellow boxes: "Managing Tomcat" (containing information about security and access to the manager webapp), "Documentation" (containing links to Tomcat 9.0 documentation, configuration, and wiki), and "Getting Help" (containing links to FAQ and mailing lists).

Step 5. Launching Jenkins

a) Running Jenkins on port 8080 (http://<Public_IP_address>:8080)



b) Fetching password and putting it in login form page

```
root@ip-172-31-33-154:/home/ubuntu# cat /var/lib/jenkins/secrets/initialAdminPassword  
410abeb955984a1ea79500b0b00e22e7
```

c) Filling required credentials

Create First Admin User

Username

Password

Confirm password

Full name

E-mail address

Jenkins 2.516.1 [Skip and continue as admin](#) [Save and Continue](#)

Step 6. Creating a Jenkins pipeline to deploy Addressbook app on Tomcat's EC2

a) Attaching maven to Jenkins (Jenkins-> manage Jenkins -> tools ->maven installations)



Jenkins

/ Manage Jenkins

/ Tools

SonarScanner for MSBuild installations

Add SonarScanner for MSBuild

SonarQube Scanner installations

Add SonarQube Scanner

Ant installations

Add Ant

Maven installations

Add Maven

Save

Apply

≡ **Maven**

Name

Maven 3

Required

☒ Install automatically ?

≡ **Install from Apache**

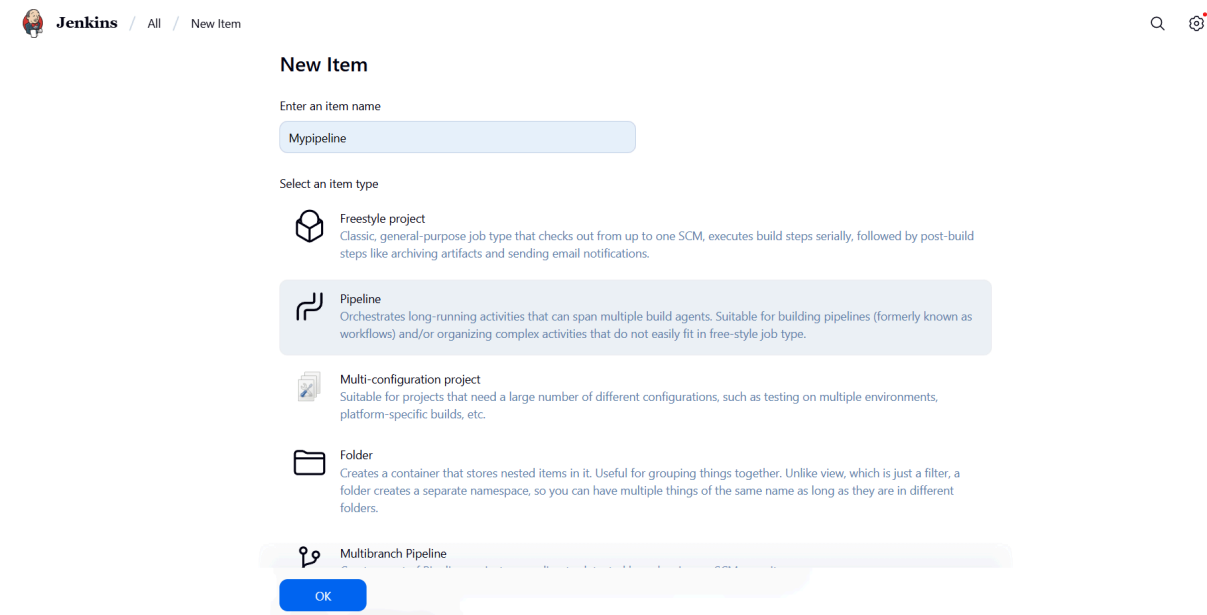
Version

3.9.11

Add Installer

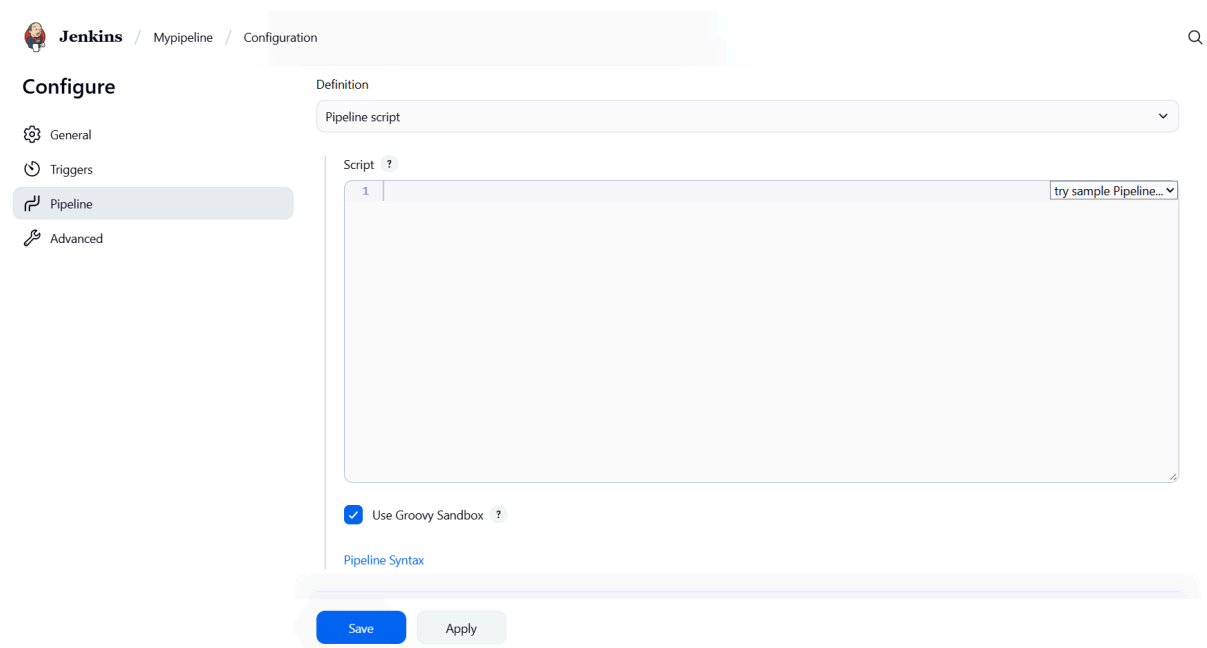
Add Maven

b) Creating a new pipeline



The screenshot shows the Jenkins 'New Item' page. At the top, the breadcrumb is 'Jenkins / All / New Item'. The main heading is 'New Item'. Below it, there is a text input field labeled 'Enter an item name' with the value 'Mypipeline'. Underneath, a section 'Select an item type' lists four options: 'Freestyle project' (Classic, general-purpose job type), 'Pipeline' (Orchestrates long-running activities), 'Multi-configuration project' (Suitable for projects that need a large number of different configurations), and 'Folder' (Creates a container that stores nested items). The 'Pipeline' option is highlighted with a blue background. At the bottom, there is a blue 'OK' button.

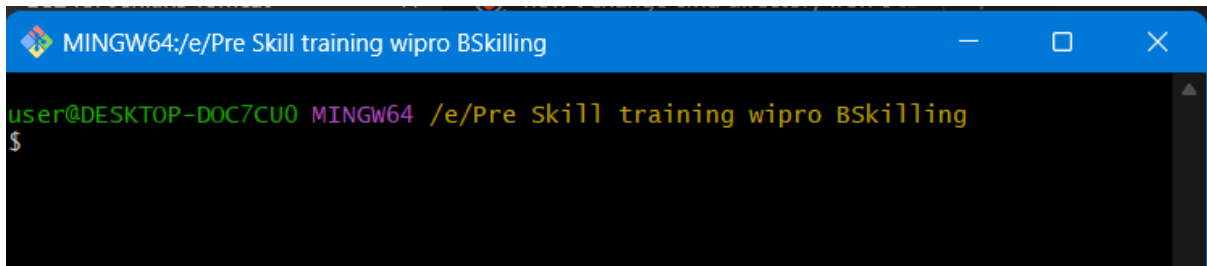
c) Writing groovy Script to perform the task



The screenshot shows the Jenkins 'Configure' page for the 'Mypipeline' item. The breadcrumb is 'Jenkins / Mypipeline / Configuration'. On the left, there is a sidebar with tabs: 'General', 'Triggers', 'Pipeline', and 'Advanced'. The 'Pipeline' tab is selected. The main area is titled 'Definition' and contains a dropdown menu set to 'Pipeline script'. Below this, there is a 'Script' section with a text area for writing the Groovy script. A checkbox labeled 'Use Groovy Sandbox' is checked. At the bottom, there are 'Save' and 'Apply' buttons.

Step 7. SSHD the Key pair to Tomcat EC2

a) Open folder where the key pair is stored then Git bash I folder

A screenshot of a Windows terminal window titled "MINGW64:/e/Pre Skill training wipro BSkilling". The prompt is "user@DESKTOP-DOC7CU0 MINGW64 /e/Pre Skill training wipro BSkilling" followed by a dollar sign "\$".

```
user@DESKTOP-DOC7CU0 MINGW64 /e/Pre Skill training wipro BSkilling
$
```

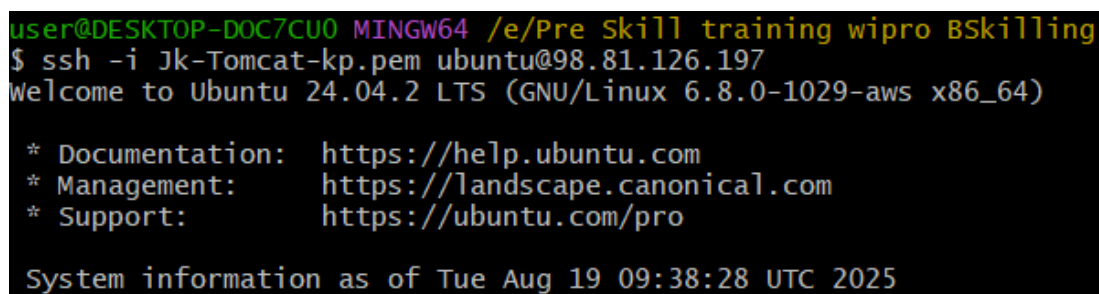
b) Provide chmod permission to the file

A screenshot of a terminal window showing the command "chmod 400 Jk-Tomcat-kp.pem" being entered at the prompt.

```
user@DESKTOP-DOC7CU0 MINGW64 /e/Pre Skill training wipro BSkilling
$ chmod 400 Jk-Tomcat-kp.pem
```

c) Run this command to add key pair to ec2

ssh -i Jk-Tomcat-kp.pem ubuntu@<public_IP_of_Tomcat_EC2>

A screenshot of a terminal window showing the output of the SSH command. It displays the Ubuntu version, documentation links, and system information.

```
user@DESKTOP-DOC7CU0 MINGW64 /e/Pre Skill training wipro BSkilling
$ ssh -i Jk-Tomcat-kp.pem ubuntu@98.81.126.197
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1029-aws x86_64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:        https://ubuntu.com/pro

System information as of Tue Aug 19 09:38:28 UTC 2025
```

Step 8. Install SSH Agent Plugin

a) Jenkins -> Manage Jenkins -> Plugins -> Available Plugins -> Search SSH Agent Plugin and install it



Plugins

📄 Updates

📦 Available plugins

⚙️ Installed plugins

⚙️ Advanced settings

🔍 Search available plugins

📦 Install



Install	Name ↓	Released	Health
<input type="checkbox"/>	Pipeline Graph Analysis 241.vc3d48fb_b_2582 Library plugins (for use by other plugins) Provides a REST API to access pipeline and pipeline run data.	2 mo 19 days ago	100
<input type="checkbox"/>	PAM Authentication 1.12 Security Adds Unix Pluggable Authentication Module (PAM) support to Jenkins	5 mo 17 days ago	97
<input type="checkbox"/>	JavaMail API 1.6.2-11		

Step 9. Create tomcat-ssh Credential

a) After SShd of EC2 do this in git bash to get private key

move out of ubuntu then (cat <key_pair_name>) copy the whole key from begin to end including both begin and end lines

```
user@DESKTOP-DOC7CU0 MINGW64 /e/Pre Skill training wipro BSkilling
$ cat Jk-Tomcat-kp.pem
-----BEGIN RSA PRIVATE KEY-----
MTTEpATBAAKCAQEFavZfv4tFv190suDSw16inO16q/whlMQm5bE7mzwqX5rahZAem
```

b) No Jenkin -> Manage Jenkin -> Credential -> Store scoped jenkins




Jenkins

/ Manage Jenkins

/ Credentials

Stores scoped to Jenkins

P	Store ↓	Domains
	System	(global)

c) Click on global and then -> Add credential then

Kind: SSH Username with private key

Username: ubuntu

ID: tomcat-ssh

Private key: enter the whole key that we just copied

then create

Global credentials (unrestricted)

+ Add Credentials



New credentials

Kind

SSH Username with private key



Scope ?

Global (Jenkins, nodes, items, all child items, etc)



ID ?

tomcat-sshh

Description ?

Username

ubuntu

☐

Treat username as secret ?

Private Key

☒

Enter directly



Private Key

☒

Enter directly

Key

Enter New Secret Below

Use this Groovy script

```
pipeline {
    agent any
    stages {
        stage('Checkout Code') {
            steps {
                git 'https://github.com/akshu20791/addressbook-cicd-project.git'
            }
        }

        stage("Compile") {
            steps {
                sh 'mvn -B -e compile'
            }
        }

        stage("Test") {
            steps {
                sh 'mvn -B test'
            }
        }

        stage("Package") {
            steps {
                sh 'mvn -B package'
            }
        }

        stage('Deploy WAR to Tomcat') {
            steps {
                sshagent(['tomcat-ssh']) {
```

```

sh ""

# Copy WAR to Tomcat webapps

scp -o StrictHostKeyChecking=no target/addressbook.war
ubuntu@98.81.126.197:/home/ubuntu/apache-tomcat-9.0.108/webapps/

# Restart Tomcat

ssh -o StrictHostKeyChecking=no ubuntu@98.81.126.197 "

    cd /home/ubuntu/apache-tomcat-9.0.108/bin &&

    ./shutdown.sh || true &&

    ./startup.sh

"

""

}

}

}

}

}

}

```

Step 10. Adding manually key.pem file

a) Since we are working on ssh we need key pair at both Jenkins EC2 and Tomcats EC2 and we have already sshd the key pair at Tomcat's EC2 now we manually add Key pair at jenkins and give it reader only permission as jenkins dosent allow to use ipen and not secure keys

Do these codes in Jenkins EC2

```
touch Jk-Tomcat-kp.pem
```

```
nano Jk-Tomcat-kp.pem
```

```

root@ip-172-31-34-39:/home/ubuntu/.ssh# touch Jk-Tomcat-kp.pem
root@ip-172-31-34-39:/home/ubuntu/.ssh# ls
Jk-Tomcat-kp.pem  authorized_keys  known_hosts
root@ip-172-31-34-39:/home/ubuntu/.ssh# nano Jk-Tomcat-kp.pem

```

Then in file add the key value of the key pair that you sshed

In the EC2 of Tomcat

Now give permission: `chmod 400 /home/ubuntu/.ssh/Jk-Tomcat-kp.pem`

this will change the file permission to read only which will not conflict with Jenkins policies

b) Now lets manually test the connection on Jenkins EC2

```
ssh -i /home/ubuntu/.ssh/Jk-Tomcat-kp.pem
```

```
ubuntu@<public_ip_of_tomcat's_EC2>
```

This should log you into Tomcat EC2 without asking for a password

Step 11. Build Pipeline

a) No click on build pipeline

The screenshot displays the Jenkins web interface for a build pipeline named 'Mypipeline', specifically for build #13. The interface is divided into a left sidebar with navigation options and a main content area. The sidebar includes links for Status, Changes, Console Output, Edit Build Information, Delete build '#13', Timings, Git Build Data, Pipeline Overview, Restart from Stage, Replay, Pipeline Steps, and Workspaces. The main content area shows the build status as 'Success' with a green checkmark. It indicates the build was started by user 'Pulkit Mathur' and provides a summary of the run's duration: 5 ms waiting, 17 sec build duration, and 17 sec total from scheduled to completion. The build is linked to a Git repository at 'https://github.com/akshu20791/addressbook-cicd-project.git' and shows the revision '0dd3c4f3fc31db20a0b2f7b31fe9453f6a12eba7'. The build is currently on the 'refs/remotes/origin/master' branch. A 'No changes' message is displayed at the bottom of the main content area. On the right side of the main content area, there are buttons for 'Add description' and 'Keep this build forever', along with a search icon, a settings icon, and a user icon. The top of the interface shows the Jenkins logo, the pipeline name 'Mypipeline', and the build number '#13'.

Step 12. Verification

a) Open Tomcat `http://<Public_IP_ofTomcat;s_EC2>:8080/addressbook/`

Filter contacts...			New contact
First Name	Last Name	Email	
George	White	george@white.com	
Daniel	Thompson	daniel@thompson.com	
Timothy	Jones	timothy@jones.com	
Peter	Wilson	peter@wilson.com	
Dan	Robinson	dan@robinson.com	
Dan	Davis	dan@davis.com	
Olivia	Davis	olivia@davis.com	
Dan	Smith	dan@smith.com	
Daniel	Anderson	daniel@anderson.com	
Alice	Thomas	alice@thomas.com	
Linda	Harris	linda@harris.com	
Daniel	Robinson	daniel@robinson.com	
Mike	Young	mike@young.com	
Umberto	Anderson	umberto@anderson.com	
Scott	Thompson	scott@thompson.com	
Rene	Martin	rene@martin.com	
Lisa	Martin	lisa@martin.com	
Peter	Martin	peter@martin.com	